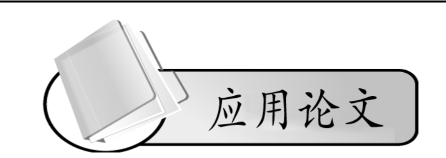


机构知识库三维模型检索与展示技术研究与实践*

吴志强　祝忠明　刘　巍　张旺强　姚晓娜
(中国科学院兰州文献情报中心　兰州　730030)

摘要: 【目的】根据机构知识库功能的扩展需求, 研究三维模型检索和展示技术, 设计并实现基于内容的三维模型检索以及 Web3D 展示与交互功能。【方法】在台湾大学开源的三维模型检索算法的基础上, 改用离屏渲染的方法获取模型的正交投影, 提取三维模型特征, 采用 Java3D 生成三维模型缩略图, 使用 Three.js 实现三维模型 Web 在线展示与交互。【结果】用户可通过提交三维模型 URL 或上传的方式, 检索三维模型, 在三维模型浏览页面可通过鼠标对模型进行旋转、缩放操作以详细查看。【局限】模型检索的结果和效率可满足当前的基本需求, 但模型的查全率和查准率还有待进一步提高, 需要持续关注三维模型检索相关技术, 优化和补充三维模型检索功能。【结论】将该模型应用于 CSpace 系统后, 可有效扩展三维模型的支持能力, 为用户提供更为多样的三维模型检索和应用方式。

关键词: 机构知识库　CSpace　三维模型检索　Web3D　Three.js
分类号: G250

1　引　言

机构知识库(Institutional Repository, IR)是科研教育机构存储、组织和管理其知识产出的重要基础设施, 随着开放获取运动和网络技术的发展, 越来越多的科研教育机构搭建起自己的机构知识库系统, 用于组织管理单位内部科研人员的科研成果, 同时根据开放共享协议, 对科研成果进行开放共享, 促进科研成果的传播, 提高科研成果的利用率, 扩大研究单位的学术影响。

目前多数机构知识库主要面向传统的期刊、会议、报告等文本格式的知识内容提供支持和服务, 对于图像、音频、视频、三维模型等非文本资源的支持能力不足[1]。三维模型作为第四代多媒体格式, 和声音、图像、视频这些多媒体数据格式相比, 在描述事物方面具有更自然、直观的特点, 同时还具有声音、图像和视频所缺少的用户交互特性。随着计算机软硬件技术、计算机图形学以及计算机视觉的发展, 三维模型在工业制造、影视制作、虚拟现实、教育、科研等诸多领域得到了广泛的应用, 已经成为一种普遍产出和利用的知识资产[2]。这要求机构知识库能够适应知识产出形式多样化的需要, 支持三维模型的存储、组织和管理的功能。

按照管理和支持能力划分, 机构知识库对三维模型的支持和管理可分为两种方式: 一种采用类似于文本内容管理的方法和技术, 支持对三维模型对象进行描述、上载和保存, 提供基于元数据描述的检索和发现利用; 另一种方式类似于图像、音视频的组

通讯作者: 吴志强, ORCID: 0000-0002-2908-6106, E-mail: wuzq@llas.ac.cn。
*本文系中国科学院 2015 年西部青年学者 B 类项目"科研过程中非文本资源的语义化组织技术研究"(项目编号: Y6AX031001)的研究成果之一。

织管理方式，提供在线浏览功能，同时通过内容分析和索引技术的应用，提供基于内容的检索、可视化导航等信息发现和利用功能。这两种组织管理方式相比，第二种方式对用户更为友好，检索和发现信息更为直观。以往受制于技术的限制，三维模型的组织和管理只能采用第一种方式，近年来随着计算机图形学和模式识别技术的发展，第二种组织管理方式成为可能。

本文对国内外三维模型检索与展示技术的发展现状进行调研和梳理，分别研究三维模型检索与展示技术，并结合中国科学院机构知识库(CSpace)的功能建设需求，采用开源的基于视觉相似性的三维模型检索算法、Web3D 展示工具 Three.js，构建了三维模型检索与展示系统。

2 相关研究

三维模型与图像、音频和视频相比，内容表现更为丰富，也具有一些特有的特征，具体表现为：由于制作工具较多，造成文件格式种类繁多，从目前比较著名的三维模型格式转换工具 3D Object Converter 所支持的三维模型格式种类来看，至少有 720 种[3]，比图像、音频和视频的格式复杂得多；完整的模型文件包括多个文件，在模型保存时一般形状、纹理贴图分开保存；常见的三维模型文件基本都可以通过文本编辑程序打开；对计算机图形处理能力的要求更高；具有平移、缩放和旋转不变性特点。由于这些特点，模型检索和展示方法多种多样，技术实现上也较为复杂。

2.1 三维模型检索

三维模型检索技术的发展始于 20 世纪 80 年代初期，经过近 40 年的发展，该领域已经积累了大量的理论研究与实践成果，也发布了一些实证检索系统。国外较有代表性的如美国普林斯顿大学形状检索与分析实验室发布的 Princeton 3D Model Search Engine[4]；德国 Konstanz 大学通用三维模型检索系统[5]；美国卡内基梅隆大学研发的基于内容的三维模型检索系统[6]等。和国外研究情况相比，国内起步较晚，但也取得了一些较有代表性的研究成果，如台湾大学 Communication and Multimedia Laboratory 发布的基于视觉相似性的在线三维模型检索系统[7]；吉林大学建立的包含基于颜色属性的三维模型库 CDB (Colored 3D Model Database)[8]，为基于颜色和文理特征的三维模型检索提供了实证平台。

在检索方式上，主要分为基于文本的检索、基于内容的检索两种方式。基于文本的检索方式，需要事先对三维模型进行相应的标记，常常无法满足用户精确查找的目的，随着计算机图形计算能力的增强，近年来基于内容的检索方式成为研究的热点。基于内容的检索，又分为通过文件检索、2D 绘图检索和手绘 3D 绘图检索三种方式。目前大多数的 3D 模型检索系统在基于内容的检索方面，主要采用文件检索方式，即通过选择要查找的 3D 模型，在模型库中查找相似的模型。

2.2 三维模型展示

三维模型展示包括模型显示和用户交互两部分，其中用户交互是模型展示的重点，包括模型的放大、缩小、旋转等功能，通过这些操作，可使用户从多角度、多层面详细观察了解模型的相关细节。展示方式主要分为 Web 在线浏览(Web3D)和客户端浏览两种：客户端浏览方式，目前支持的软件较多，用户只需要安装相应的软件即可；对于 Web3D，主要的实现技术或工具有云渲染、Java3D[9]、Away3D[10]、Unity Web Player 3D[11]、Cult3D[12]、Three.js[13]等。

基于云渲染技术的 Web3D 是将三维模型的渲染过程放到云端进行，客户端不需要任何的设置，在各种平台，如手机、平板电脑、PC 上都能展示。在三维模型展示过程中，本地客户端不加载模型文件，只加载渲染后的图片，没有模型加载的过程，用户不需要长时间的等待，同时根据云端的配置理论上可以展示任何大小的模型文件，安全性也较高。

Java3D 主要通过 Java Apple 实现 Web3D，客户端需要安装 JRE(Java Runtime Environment)，且需要对浏览器的安全选项进行设置，用户操作较为麻烦；Away3D、Unity Web Player 3D、Cult3D 这些工具都需要用户安装插件; Three.js 是 WebGL 的开源框架，可作为普通的 JS 文件在 Web 中调用，使用 Three.js 需要浏览器支持 WebGL，目前支持 WebGL 的浏览器很多，例如 Chrome、FireFox、360 安全浏览器 6.0、IE11 等，一些老的浏览器不支持 WebGL，但支持 WebGL 是浏览器未来发展的方向。

通过对相关三维模型检索和展示技术的梳理以及

实验研究发现：虽然已发布的三维模型检索系统和平台较多，但大多不提供三维模型检索的开放调用接口，也没有相关算法的详细说明，无法有效整合应用到机构知识库系统中。台湾大学发布的基于视觉相似性的三维模型检索算法，公开了程序源码，实验结果表明，具有较好的三维模型查全率和查准率，同时在现有的源码基础上，可根据机构知识库系统的需求，进行修改和接口定制，便于机构知识库系统的调用，因此，选择台湾大学发布的三维模型检索算法进行机构知识库三维模型检索研究。

对于三维模型展示技术，从用户体验的角度出发，无插件、无需用户配置的 Web3D 展示与交互方式最为友好，用户也最易接受。符合这个条件的主要有 Three.js 和云渲染技术，云渲染需要硬件和软件双重支持，对硬件和软件的投入都较大，Three.js 已封装实现了很多底层的 Web3D 功能，使用时根据其提供的 API 文档，在 Web 页面引入相关的 JS 文件，即可调用相关的算法，同时也支持 PC、平板电脑、手机等多终端展示，符合 CSpace 机构知识库系统前端界面自适应需求，因此采用 Three.js 进行机构知识库三维模型展示功能研究与实现。

3 机构知识库三维模型检索与展示系统功能框架

机构知识库三维模型检索与展示系统功能框架如图 1 所示，分为两个部分，检索部分主要包括模型格式转换、特征提取、索引、检索；Web3D 展示包括模型缩略图、模型显示、交互操作功能。

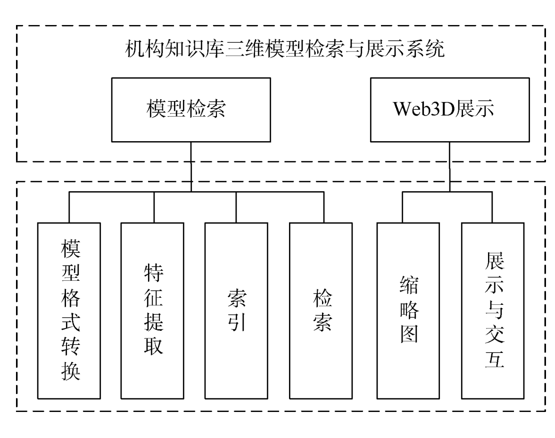


图 1 系统功能框架

(1) 模型格式转换：三维模型格式多样，若不将模型格式转换统一，后期的特征提取、缩略图、模型展示功能的实现存在非常大的格式兼容性困难。为了后期处理的方便，当用户提交三维模型时，首先判断其格式是否为设定的格式，若不是，则进行格式转换。

(2) 特征提取：三维模型具有丰富的表现力，可利用的模型特征多样，如形状、轮廓、颜色等，选择一定的模型参数表征模型，进行特征提取，为后期的相似性计算及检索做好数据准备。

(3) 索引：选择索引存储方式，设计索引机制，构建和维护三维模型索引，当用户添加或删除三维模型时，进行相应模型索引的添加或删除操作，使索引和三维模型库保持一致。

(4) 检索：提取用户提交的三维模型特征，根据索引列表，分别计算与模型库中的三维模型的相似度，设置模型相似度阈值，返回根据相似度大小排序的相似模型列表。

(5) 缩略图：为便于用户进一步查找所需要的模型，在模型检索结果页面，根据相似度大小，展示模型的二维缩略图，若用户想进一步了解某一三维模型，可点击缩略图进入模型展示页面进行查看。

(6) 模型展示与交互：在 Web 页面三维模型展示区，用户可通过鼠标进行模型的旋转、缩放等操作，详细查看模型。

4 关键功能的设计与实现

4.1 三维模型处理

机构知识库三维模型检索与展示系统实现过程中，需对三维模型进行一些处理，如格式转化、缩略图、特征提取，这些处理过程所使用的工具或所依赖的工具包对系统底层环境有一定的要求。若将这些三维模型处理实现方法嵌入到现有的机构知识库系统中，将对现有机构知识库系统的操作系统兼容性产生影响，同时在机构知识库安装或升级过程中，需要人工配置和调试三维模型处理方法，给系统维护带来不便，因此，在机构知识库三维模型检索与展示系统实现过程中，采用 Java RMI(Remote Method Invocation)[14]，将三维模型处理过程配置到一台 Windows Server 2003 服务器上，机构知识库系统可通过 IP 地址和端口进行远程调用，具体过程如图 2 所示。

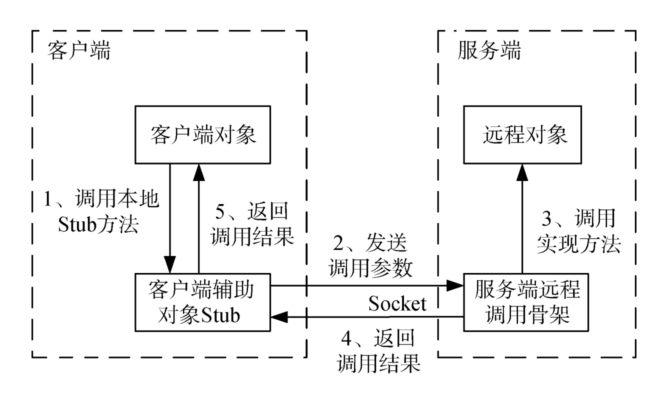


图 2　三维模型处理方法调用过程

客户端首先连接三维模型处理服务器，然后通过远程调用方法上传三维模型，模型传输成功后，根据客户端传递的处理参数，服务端完成三维模型格式转换、特征提取、缩略图生成，并将得到的特征参数文件、缩略图、转换后的模型文件以及处理结果返回给客户端。机构知识库调用三维模型处理过程核心代码如下所示:

```
// 连接三维模型处理服务器
if (!connect())
   return "Can not connect the 3D server!";
// 上传三维模型
if (!objFeatureGenerate.upload(fileName, fileToByte(srcFile)))
   return "Upload 3D file failed!";
// 模型转换
if (isconvert2obj) {
   if (!objFeatureGenerate.convert2obj(fileName))
      return "convert2obj 3D file failed!";
}
if (objFeatureGenerate.isFindObj (fileNameWithoutExt+".obj")) {
// 产生三维模型缩略图
   if (isGetImage) {
      if (!objFeatureGenerate.generateImage (fileNameWithoutExt))
         return "Generate 3D image failed!";
   }
   // 产生三维模型特征参数文档
   if (isGenerateFeature) {
      if (!objFeatureGenerate.generateFeature (fileNameWithoutExt))
         return "Generate 3D feature failed!";
   }
// 下载三维模型特征文档
   try {
      if (isGetImage && objFeatureGenerate.
       isFindObj(fileNameWithoutExt+".jpg"))
         save(fileNameWithoutExt + ".jpg");
   if (isconvert2obj) {
      if(isSaveObj && objFeatureGenerate.
       isFindObj(fileNameWithoutExt+".obj"))
         save(fileNameWithoutExt + ".obj");
   else
      objFeatureGenerate.deleteFile(fileNameWithoutExt + ".obj");
   }else{
       objFeatureGenerate.deleteFile(fileName);
   }
   if (isGenerateFeature) {
      for (String ex : ext) {
         if(objFeatureGenerate.isFindObj(fileNameWithoutExt+"."
            + ex))
            save(fileNameWithoutExt + "." + ex);
      }
……
```

机构知识库调用三维模型处理的方法类如图 3 所示，服务端提供调用的方法如图 4 所示，机构知识库中三维模型处理接口所在的 Package 和服务器端三维模型处理方法所在的 Package 保持一致，否则方法调用时，将会出现找不到调用方法的错误。

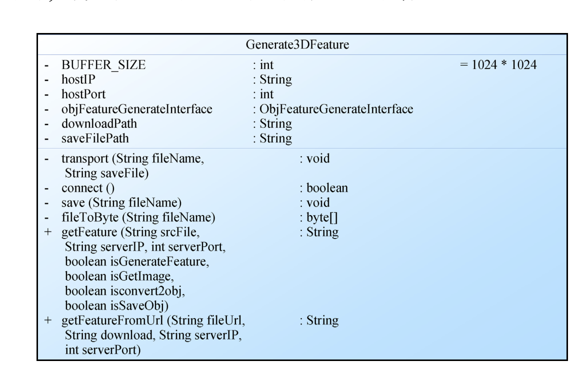


图 3　三维模型客户端调用类

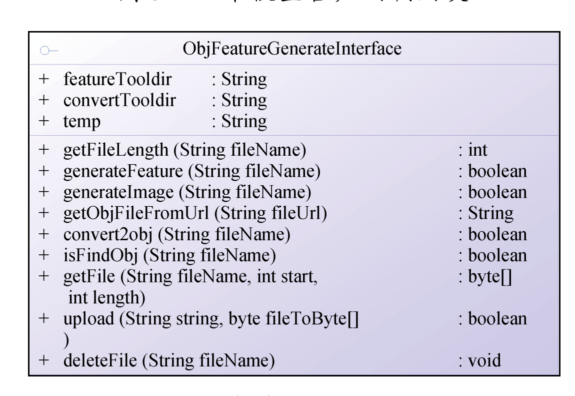


图 4　三维模型处理方法接口

三维模型处理部分主要实现以下三个功能:

(1) 三维模型格式转换：三维模型的格式类型繁多，为便于后期模型检索和展示功能的实现，采用目前应用广泛、格式转换较为方便的 obj 格式作为统一的三维模型格式，格式转换所用软件为 3D Object

Converter，支持常见三维模型格式如：3ds、dxf、cad、geo、stl、c4d、rwx 向 obj 格式的转换，软件安装在三维模型处理服务器上，通过方法 convert2obj 进行调用。

(2) 缩略图生成：三维模型文件为文本格式，要生成模型缩略图，首先读取三维模型文件，渲染并保存到一定的容器中，最后提取容器中的图像数据流并保存。这一过程中，使用 Java3D 读取文件，采用离屏渲染方法完成渲染工作，读取内存中的缩略图数据保存为 JPG 格式的文件。由于三维模型的纹理信息加载过程较复杂，模型加载的时候，未加载纹理文件，缩略图也不包含模型的纹理信息。生成的三维模型缩略图作为三维模型的子条目存储在机构知识库系统中。

(3) 模型特征提取：所提取的三维模型特征和使用的检索算法直接相关，现有的三维模型算法较多，通过实验，模型检索采用台湾大学发布的基于视觉相似性的三维模型检索算法[15]，其算法认为，若两个三维模型相似，则从所有角度看，都是相似的，因此，在光场下，三维模型多个角度相似性大小之和就可以衡量两个模型的相似性，所采用的三维模型特征为一系列的光场描述符号(LightField Descriptor)。原有的算法，在特征提取过程中，模型渲染后的图像首先存储在 Glut Windows 中，再从界面中截取图像。若 Glut 界面被遮挡，则提取图像有残缺，会影响到后面模型特征提取的结果，因此将三维模型渲染方法修改为离屏渲染，并修改其参数调用方法，便于 Java 参数调用。

4.2 三维模型索引与检索

三维模型索引使用 Solr-4.10.2[16]进行维护，索引的构建和检索算法是对应的，前述三维模型检索算法，在模型检索过程中，模型列表存放在一个文本文件中，通过文件路径进行调用，模型列表每行表示一个三维模型特征存储路径，如 Example/m0，模型特征在模型比较时以文件的方式进行调用。因此，Solr 索引文件主要存放三维模型特征的路径信息，Solr 的索引配置文件 schema.xml 中定义的索引字段和索引类型如下：

```
<fieldtype name="string" class="solr.StrField" sortMissingLast=
  "true" omitNorms="true"/>
<field name="id" type="string" indexed="true" stored="true"
  multiValued="false" required="true"/>
<field name="_version_" type="long" indexed="true"
  stored="true"/>
```

三维模型文件在机构知识库中以 Bitstream 对象进行存储，模型特征文件的名称以三维模型的 Bitstream ID 命名。向机构知识库系统提交三维模型时，系统根据三维模型文件的后缀名，判断文件类型，若为三维模型文件，则触发模型格式转换、特征提取、缩略图提取、Solr 索引建立等程序，具体的 Bistream 创建、修改和删除时 Solr 索引机制已在文献[17]进行了详细阐述，不再赘述。

三维模型检索过程如图 5 所示，模型检索时，首先判断用户提交的模型文件的格式，如果为 obj 格式则直接进行特征提取，如果非 obj 格式，则首先进行格式转换；然后和机构知识库中所有的三维模型进行特征比较，分别计算相似度；最后排序输出检索结果。

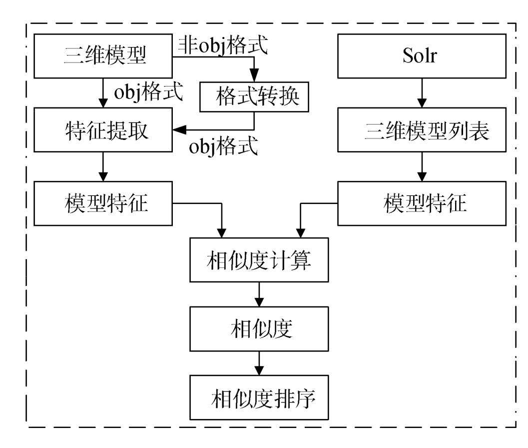


图 5 三维模型检索过程

三维模型比较后，返回的相似度大小以正整数表示，数值为零时，表示两个三维模型为同一模型，数值越大，相似度越小。为向用户返回需要的三维模型，设置三维模型相似度阈值，根据实验，将阈值暂时设置为 1000，大于 1000 的从相似度列表中删除。根据返回的三维模型相似度列表，在检索结果页面通过三维模型缩略图分别展现可能的三维模型和相似的三维模型。当相似的三维模型较多时，为提高检索结果页面的加载速度，对检索结果进行分页显示，通过 Ajax 异步方式在原有页面上动态加载下一页检索结果。

4.3 模型展示与交互

Three.js 是 WebGL 的开源框架，已封装实现了很多 Web3D 的功能，使用其提供的 obj 格式三维模型展示示例方法即可实现简单的三维模型展示功能，但加载的三维模型无法根据画布大小自适应模型大小和位置，造成有些较小的模型在画布上看不见，较大的模

型又超出整个画布的范围，同时也无法做到模型在画布的中心显示。对于这一问题，在实现机构知识库三维模型展示功能时，在示例程序的基础上，使用 Three.js 提供的 Box3 函数设置模型居中显示，并通过 Box3 函数计算模型的大小，然后根据模型大小，设置 camera 的 z 轴参数解决了画布中模型显示比例问题。模型的旋转、缩放这些交互操作的实现，通过调用 OrbitControls.js 实现。

机构知识库三维模型展示功能嵌入于 display-item.jsp 页面，模型渲染时，不加载纹理文件，设置画布的颜色为白色，灯光为蓝绿色，渲染后的三维模型效果在页面的非文本展示区呈现，display-item.jsp 可加载多个三维模型文件，为了便于调用，构建 function displayModel (container, objname)，container 为渲染后三维模型显示容器，objname 为三维模型文件路径，模型加载和渲染方法参照 Three.js 示例程序完成。所加入的三维模型根据画布大小自适应模型大小和位置的程序代码如下：

```
var box3 =new THREE.Box3 ();
box3.setFromObject(object)
box3.center(object.position );
object.position.multiplyScalar( - 1 );
var a = new THREE.Vector3();
a=box3.size();
var objectSize = Math.max(a.x, a.y)
var fov = camera.fov * ( Math.PI / 180 );
var distance = Math.abs( objectSize / Math.sin( fov / 2 ) );
camera.position.z=distance;
```

4.4 实现效果

为了验证设计实现的三维模型检索和展示功能的效果，首先从台湾大学 Communication and Multimedia Laboratory 发布的基于视觉相似性的在线三维模型检索系统下载了 20 多个 obj 格式的三维模型文件，使用 3D Object Converter 将其中部分文件转换为 3ds、dxf、cad、geo、stl、c4d、rwx 格式，提交到机构知识库系统中，建立三维模型检索库。在提交过程中系统会自动根据设置的三维模型文件后缀对提交的文件进行判断，若是三维模型文件，但不是 obj 格式，则触发三维模型格式转换程序，将文件格式转换为 obj 格式，然后系统会自动完成三维模型特征提取、缩略图生成、索引构建等工作。

在页面的检索功能区，点击照相机图标，在弹出的检索窗口上传三维模型文件，系统根据文件的后缀名判断是否为三维模型，若是，则进行三维模型检索。检索结果表明，前期上传到系统中的不同格式的三维模型文件，都能够按照设定的处理规则，自动完成模型转换、缩略图生成、特征提取及索引工作，检索结果根据相似度大小排序，采用缩略图的方式呈现，根据设置的相同模型和相似模型阈值大小，可展示最可能的模型以及其他相似模型。检索结果页面如图 6 所示。



图 6 三维模型检索结果页面

根据检索结果，点击相应的缩略图，可进入模型浏览页面进行详细查看，并可通过鼠标进行模型的旋转、缩放操作，模型展示如图 7 所示。

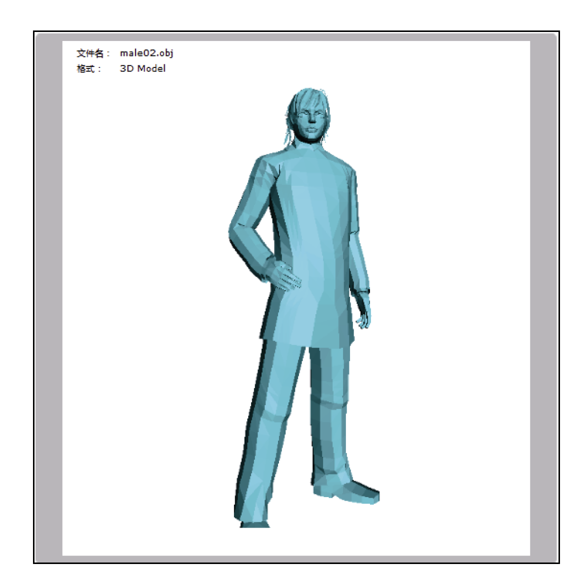


图 7 三维模型展示与交互

5 结　语

机构知识库是管理机构知识产出的重要基础设施，随着信息技术的发展，知识产出的格式也趋于多样化，为了扩展现有机构知识库的知识管理和支持能力，本文结合CSpace系统对三维模型检索和展示功能的需求，使用开源的三维模型检索算法，构建机构知识库三维模型检索系统，并采用 Three.js 实现了三维模型在 Web 界面的展示与交互功能。经测试，模型检索的结果和效率可满足当前的基本需求，但查全率和查准率还有待提高，未来需要持续关注三维模型检索方面的最新进展，将三维模型检索的新技术应用于机构知识库功能扩展上，同时还需对已有的系统进一步优化，在基于文本和内容的检索功能基础上，研究语义化检索方法，为用户提供更好的机构知识库三维模型支持服务。

参考文献:

[1] 张晓林. 机构知识库的发展趋势与挑战[J]. 现代图书情报技术, 2014(2): 1-7. (Zhang Xiaolin. Trends and Challenges for Institutional Repositories [J]. New Technology of Library and Information Service, 2014(2): 1-7.)

[2] 路通. 三维 CAD 模型检索综述[J]. 计算机科学, 2012, 39(4): 14-22, 27. (Lu Tong. Retrieval of 3D CAD Model: Survey [J]. Computer Science, 2012, 39(4): 14-22, 27.)

[3] 3D Object Converter [EB/OL]. [2016-06-10]. http://3doc.i3dconverter.com/features.html.

[4] Princeton 3D Model Search Engine [EB/OL]. [2016-06-10]. http://shape.cs.princeton.edu/search.html.

[5] Zarpalas D, Kordelas G, Daras P. Recognizing 3D Objects in Cluttered Scenes Using Projection Images [C]//Proceedings of the 18th IEEE International Conference on Image Processing. 2011: 673-676.

[6] AMP 3D Model Retrieval [EB/OL]. [2016-06-10]. http://chenlab.ece.cornell.edu/projects/3DModelRetrieval/.

[7] 3D Model Retrieval System Based on LightField Descriptors [EB/OL]. [2016-06-10]. http://3d.csie.ntu.edu.tw/.

[8] 程言志, 吕天阳, 王森, 等. 基于表面颜色属性的三维模型检索[C]//NDBC2009 第 26 届中国数据库学术会议论文集. 2009: 366-372. (Cheng Yanzhi, Lü Tianyang, Wang Sen, et al. 3D Model Retrieval Based on Surface Color Properties [C]//Proceedings of NDBC2009. 2009: 366-372.)

[9] Java3D [EB/OL]. [2016-03-20]. https://java3d.java.net/binary-builds.html.

[10] Away3D [EB/OL]. [2016-04-10]. http://away3d.com/.

[11] Unity3D [EB/OL]. [2016-04-10]. http://unity3d.com/cn/webplayer/.

[12] Cult3D [EB/OL]. [2016-04-10]. http://www.web3d.com.cn/new/teach/cult3d/.

[13] Three.js [EB/OL]. [2016-04-10]. http://threejs.org/.

[14] RMI [EB/OL]. [2016-04-10]. https://docs.oracle.com/javase/tutorial/rmi/.

[15] Chen D, Tian X, Shen Y, et al. On Visual Similarity Based 3D Model Retrieval [C]//Proceedings of Computer Graphics Forum. Blackwell Publishing, Inc, 2003: 223-232.

[16] Solr [EB/OL]. [2016-04-10]. http://lucene.apache.org/solr/.

[17] 吴志强, 祝忠明, 刘巍, 等. 基于 LireSolr 的机构知识库图像检索[J]. 图书馆学研究, 2016(14): 58-63, 39. (Wu Zhiqiang, Zhu Zhongming, Liu Wei, et al. The Image Retrieval of LireSolr-Based for Institutional Repository [J]. Research on Library Science, 2016(14): 58-63, 39.)

作者贡献声明:

吴志强: 论文起草, 系统功能设计与实现;

祝忠明: 提出研究思路, 设计研究方案;

刘巍: 系统功能部署与测试;

张旺强: 三维模型索引接口实现, 论文修改修订;

姚晓娜: 三维模型检索技术调研。

利益冲突声明:

所有作者声明不存在利益冲突关系。

支撑数据:

支撑数据由作者自存储, E-mail: wuzq@llas.ac.cn。

[1] 吴志强. GenerateFeature.zip. 三维模型远端处理程序.

收稿日期: 2016-08-25
收修改稿日期: 2016-09-26

Retrieving 3D Models from Institutional Repository

Wu Zhiqiang Zhu Zhongming Liu Wei Zhang Wangqiang Yao Xiaona
(Lanzhou Library, Chinese Academy of Sciences, Lanzhou 730030, China)

Abstract: [**Objective**] This paper aims to explore new content-based technology to retrieve and display 3D models, and expands the services of institutional repository. [**Methods**] First, we modified the open source 3D model retrieval algorithm created by the Taiwan University. Second, we obtained the orthogonal projection and features of the 3D models with the off-screen rendering technique. Finally, we used Java3D technique to generate the thumbnails of the 3D models, and then presented them online with the help of Three.js. [**Results**] We could retrieve 3D models from the institutional repository by submitting their URLs or the uploading methods. User could also use mouse to rotate or zoom in/out the 3D models while browsing them online. [**Limitations**] The proposed 3D model retrieval technique met needs of the institutional repositories. However, the recall and precision of the new system could to be improved with the help of the latest techniques in 3D model retrieval. [**Conclusions**] The proposed method helps the CSpace system manage 3D model collections effectively, which provides more options to retrieve and use the 3D models.

Keywords: Institutional Repository CSpace 3D Model Retrieval Web3D Three.js

HathiTrust 研究中心发布要素数据集 1.0 版本

HathiTrust 于近日发布一个开放数据集: HathiTrust 研究中心(HathiTrust Research Center, HTRC)抽取的要素(Extracted Features, EF)数据集 1.0 版本。该数据集为研究人员提供了从 HathiTrust 数字图书馆(HathiTrust Digital Library, HTDL)全文抽取获得数据的开放获取。

EF 数据集为学术界打开了探索完整 HathiTrust 馆藏的大门, 学者们可以深入研究 HathiTrust 馆藏的历史和文化趋势, 语料库中主题的兴起和衰落情况, 以及从 16 世纪到 20 世纪末的出版物中单词和书写结构的演变情况。 EF 数据集提供关于 HathiTrust 数字图书馆中每一卷每一页的字、行、句子成分, 以及其他细节的量化信息。 此外, EF 数据集还允许研究人员深入分析给定部分卷的内容。

EF 数据集的数据从 HathiTrust 数字图书馆的 1370 万卷的 5 亿个页面的 2 万亿个单词中抽取而得。EF 数据集的初始版本是从 HathiTrust 公共馆藏中抽取, 目前已经成功为经济学、历史学、语言学、文学和社会学等领域的学者提供了新颖的研究资料。

“抽取的要素数据集为学术界创造了前所未有的科研和教学机会,” HathiTrust 研究中心联合主任, 伊利诺伊大学香槟分校信息科学学院教授 J. Stephen Downie 表示: “期待看到学术界如何在科学研究、实验室和教室中利用 EF 数据集。”

HathiTrust 的执行董事 Michael Furlough 说: “我们成立 HathiTrust 研究中心的目的是帮助研究人员完全挖掘 HathiTrust 馆藏。EF 数据集的发布通过从整个语料库抽取相关的要素数据, 为研究人员提供了一种新颖而有效的方式来挖掘 HathiTrust 馆藏。”

(编译自: https://librarytechnology.org/news/pr.pl?id=22069)

(本刊讯)